UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC H3C 3J7

A GENERAL EXCHANGE PROCEDURE FOR THE MTSP

by

Guy Lapalme*
Jean-Yves Potvin*
Jean-Marc Rousseau**

* Département d'informatique et de recherche opérationnelle, Université de Montréal.

** Centre de recherche sur les transports, Université de Montréal.

Université de Montréal
Centre de recherche sur les transports - Publication #489
Septembre 1986

Dépôt légal - 3e trimestre 1986

A General exchange procedure for the MTSP

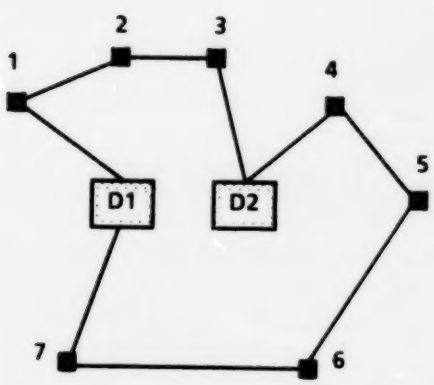
Abstract This paper introduces a technique generalizing the k-opt exchange procedure for the MTSP (Multiple Traveling Salesmen Problem). More precisely, we consider exchanges leading to the partition of a tour into a number of smaller subtours. These subtours can then be recombined in order to obtain an equivalent single tour. By this mean, we consider more opportunities for exchange and improvement of the solution. Computational tests are presented comparing the classical 2-opt and 3-opt exchange procedures with a general 2-opt exchange procedure

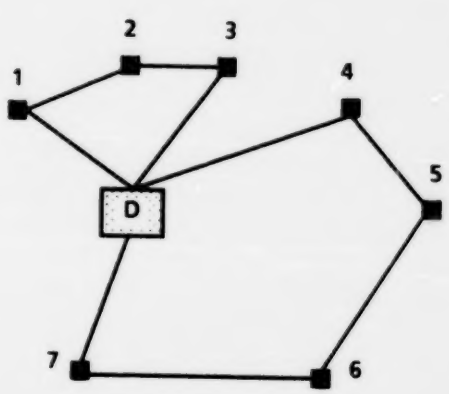
Section 1. Introduction

It is well known that the apparent generalization that the MTSP provides is somewhat illusory, since each problem of this type can be converted into an equivalent TSP [1,5]. The equivalence is obtained by creating M copies of the depot D1,...,DM, each connected to the other nodes exactly as was the original depot (with the same lengths). The M copies are not connected or are connected by arcs each with a length sufficiently large to prevent their use. These modifications define a new "augmented" network in which any MTSP solution appears as a unique tour (see figure 1a).

Assuming a symmetric network, it is now possible to apply the classical k-opt exchange procedures on this unique tour [2,3]. As we can see in figure 1b), some exchanges will actually shift nodes from one route to another. Our goal here is to generalize this approach in order to allow the partition of the tour into subtours. For a TSP the partition leads necessarily to a non feasible solution, but for a MTSP some of these solutions are feasible. In fact, a solution of this kind is feasible iff at least one copy of the depot is included in each of the subtours. This observation has not been reported in the current litterature and our analysis shows it's interest.

a)





b) Exchange of edges {(1, 2) (7, D1)} for edges {(D1, 2) (7, 1)}

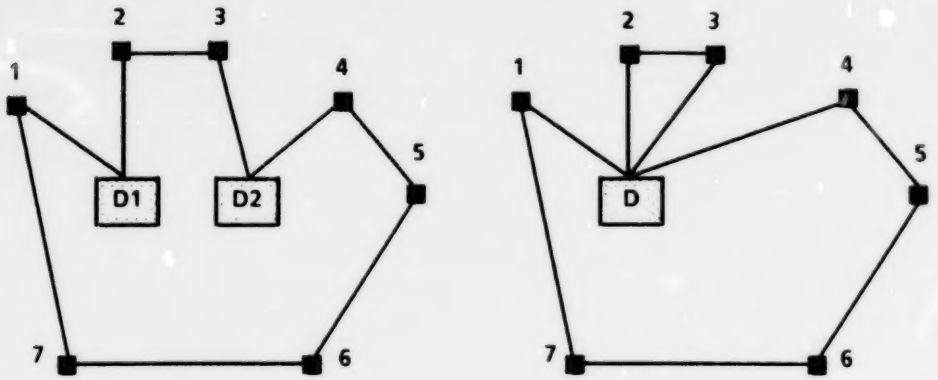


Fig. 1 Equivalence 2TSP - TSP

Section 2 The general exchange procedure

We can see in figure 2 that there exists three ways to link two subchains produced by the deletion of two edges in a single tour. Leaving out case (a) which restores the initial tour, a classical 2-opt exchange considers only case (b). Likewise, in figure 3, we can see that there exists fifteen ways to link three subchains and a classical 3-opt exchange considers only cases (b) to (h).

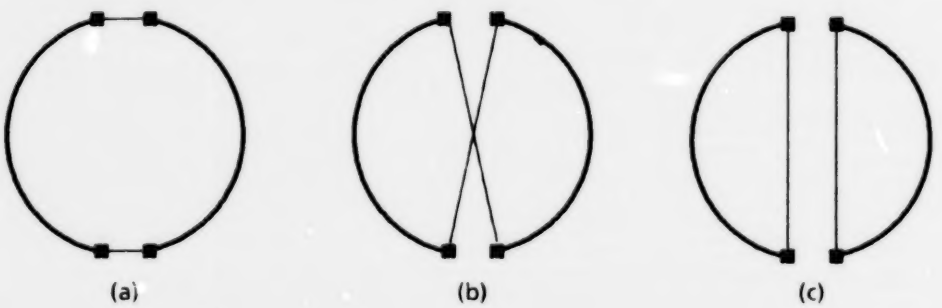


Fig. 2 Connecting two subchains

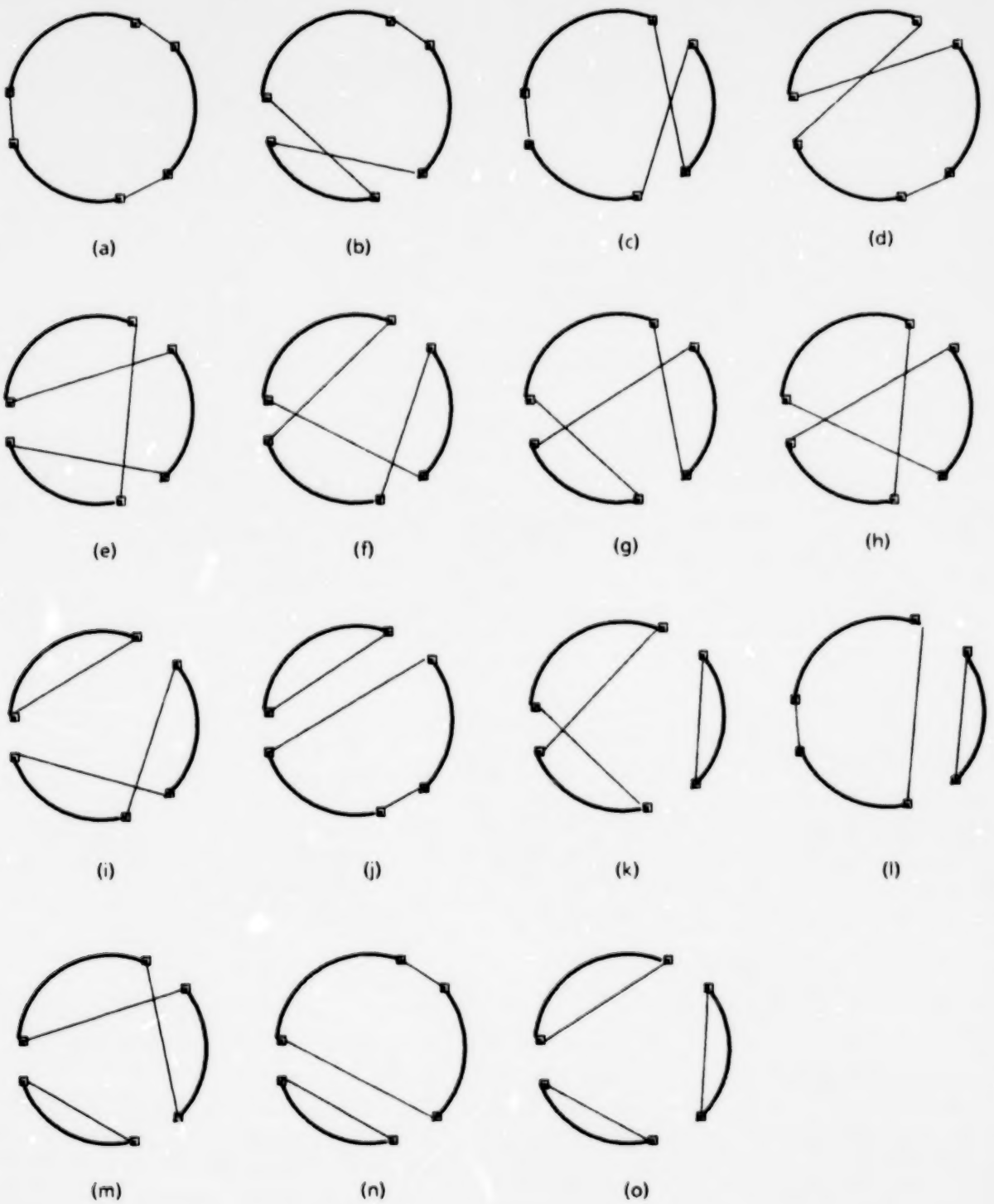


Fig. 3 Connecting three subchains

When dealing with a MTSP, some of the connections generating multiple subtours are worth some considerations. In fact, for a 2-opt exchange, we note the followings:

a) if at least one copy of the depot is included in each of the two subchains, then all types of connections described in figure 2 lead to a feasible solution

b) if a subchain is free of any copy of the depot, then only cases (a) and (b) must be considered

Similarly, for a 3-opt exchange, we get:

a) if at least one copy of the depot is included in each of the three subchains, then all types of connections described in figure 3 lead to a feasible solution

b) if one subchain is free of any copy of the depot, then only twelve types of connections must be considered. The non authorized types implies the closure on itself of the "copy free" subchain, thus forming a "copy free" subtour.

c) if two subchains are free of any copy of the depot, then only cases (a) to (h) must be considered.

Hence, the general exchange procedure retains all the exchanges leading to a feasible solution, even those generating multiple subtours. In the case of a 2-opt or 3-opt exchange procedure, it will at most double the number of exchanges considered. We must note finally that for any solution with multiple subtours, there exists an equivalent unique tour obtained by the judicious exchange of some equivalent edges (i e. edges connecting a given node to each copy of the depot, as depicted in figure 4).

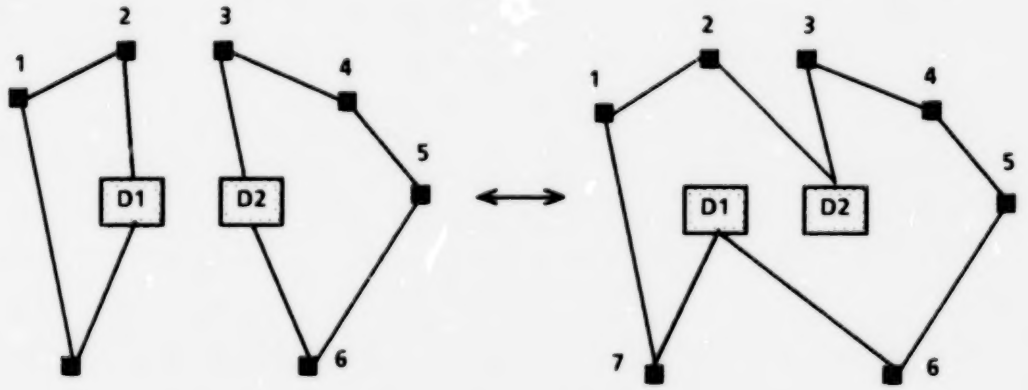
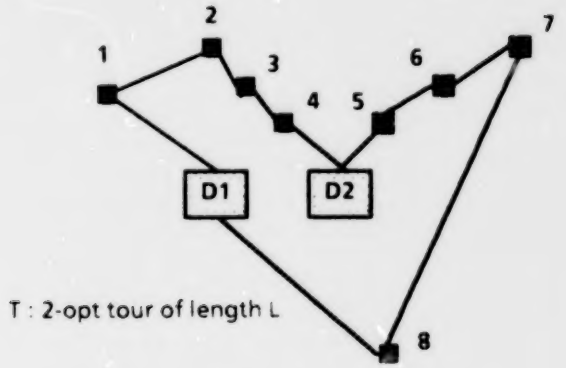


Fig. 4 Generating a unique tour

Following the terminology of Lin [2], a tour is said to be k-opt* if it's length can not be lowered by a general k-opt exchange. On this premise, we can easily infer the following result:

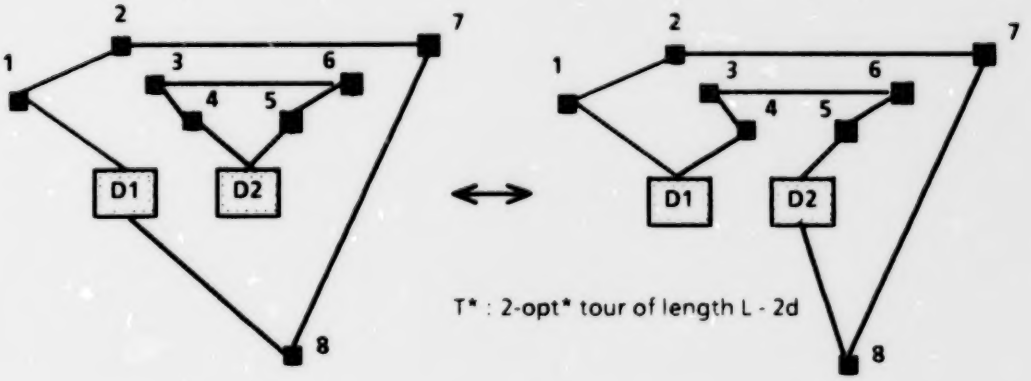
(2.1) Any k-opt* tour is k-opt

As we can see in figure 5, the inverse is not true.



T : 2-opt tour of length L

Exchange of edges {(2,3) (6,7)} for edges {(2,7) (6,3)}



T* : 2-opt* tour of length L - 2d

$d(i,j) = x \qquad \forall (i,j) \in E$

$d(i,j) = x + 2d \qquad \forall (i,j) \notin E,\ (i,j) \neq (2,7), (3,6)$

$d(2,7) = d(3,6) = x - d$

where $E = \{\text{edges on } T\} \cup \{(4,D1)\ (5,D1)\ (1,D2)\ (8,D2)\}$

$d > 0$

Fig. 5 2-opt versus 2-opt* tour

In figure 5, we can see that four edges have been exchanged in order to obtain the 2-opt* tour (including the exchange of equivalent edges to reunify the subtours into a unique tour). In fact:

(2.2) A general k-opt exchange can lead to the substitution of at most 2k edges in a tour defined on the augmented network. Moreover, if there are k + x ($x > 0$) substitutions then at least x of them involve equivalent edges.

Hence, the general k-opt exchange procedure is more powerful than the classical approach since the latter can authorize no more than k substitutions in a given tour.

Section 3 An iterative algorithm

Obviously, we can embody the general k-opt exchange procedure in an iterative algorithm like:

1) Generate an initial solution.

2) Try to improve the length of the tour using the general k-opt exchange procedure. If the exchange leads to multiple subtours, generate an equivalent unique tour.

3) Repeat step 2 until no more improvement is attainable.

The complexity of step 2 of this algorithm is as follows:

$$\binom{n}{k} (2^{k-1}(k-1/2)! - 1)$$

where n = number of edges in the tour

$(k-1/2)! = (k-1/2) \times (k-3/2) \times \ldots \times 3/2$

The equivalent algorithm designed for the classical k-opt exchange procedure gives:

$$\binom{n}{k} (2^{k-1}(k-1)! - 1)$$

We must emphasize that for a given k ($k > 1$), the complexity of the general and classical approaches are respectively of order

$$C'\binom{n}{k} \quad \text{and} \quad C\binom{n}{k} \quad \text{where } C' > C.$$

Thereby, the complexity is in the order of n^k for both approaches.

For example, with $k = 2$ we get an order of n^2 with $C' = 2$ and $C = 1$. Thus, the number of computations needed by the general exchange procedure is just two times the number needed by the classical exchange procedure, no matter of the size of the tour (the number of edges in the tour). This fact seems rather interesting, since the classical 3-opt exchange procedure requires on the order of n^3 computations.

Section 4. Some experiments

To illustrate the interest of the general approach, we proceed to some experiments using randomly generated networks with 20, 40, 60 and 80 nodes (ten networks for each size). We generate initial tours for two and three salesmen with a fairly good heuristic known as "farthest insertion" [4]. We then applied the 2-opt, 2-opt* (general 2-opt) and 3-opt exchange algorithms in order to compare their performance. In Table I, we present for each network's size and for each kind of algorithm the total length of the tours generated and the percentage of improvement over the initial solution (INIT). Each result shown is the mean taken over ten problems. The computation times are not shown and are not relevant due to the relationship between the time complexity of the algorithms used, that is: if TIME(opt2) = x and N = number of edges in the tour then TIME(opt2*) = 2x and TIME(opt3) = Nx.

These results show the superiority of the general 2-opt exchange procedure over a classical 2-opt. However, in about 60% of the problems, the two methods give the same solution. It means that the general approach does not guarantee at all that a better solution will be found, but it doesn't cost much to try it and it could provide great benefits. Generally speaking, the general 2-opt exchange procedure can be viewed as a good compromise between low computation time and high solution's quality, that is, a good compromise between a classical 2-opt and a classical 3-opt.

2TSP	20	%	40	%	60	%	80	%
INIT	2253 65		2895 81		3257 47		3874 01	
2OPT	2194 64	2 6	2828 13	2 3	3185 81	2 2	3777 15	2 5
2OPT*	2159 38	4 2	2803 14	3 2	3156 49	3 1	3746 66	3 3
3OPT	2088 85	7 3	2742 81	5 2	3071 79	5 7	3641 44	6 0

3TSP	20	%	40	%	60	%	80	%
INIT	2500.17		3133.71		3488.88		4082.60	
2OPT	2338.31	6.5	2965 38	5.4	3338.86	4.3	3878.047	5.0
2OPT*	2283.96	8 6	2916 28	6.9	3286.53	5 8	3804 98	6.8
3OPT	2219 07	11.2	2837 22	9 5	3167.90	9 2	3643 00	10.8

Table I Computational results

Section 5 Conclusion

We devise a technique generalizing the classical k-opt exchange procedure for the MTSP. The complexity of the general and classical approaches are both of order n^k, where n is the number of edges in the tour. Our experiments show the superiority of the general approach for $k = 2$. Since the number of computations needed for a general 2-opt exchange is just two times the number needed for the classical 2-opt and is independent of the number of edges in the tour, it seems to be a good compromise between low computation time and high solution's quality.

References

[1] Bellmore A. & Hong S., "Transformation of multi-salesman problem to the standard traveling salesman problem", J. ACM 21, p. 500-504 (1974).

[2] Lin S. , "Computer solutions of the traveling salesman problem", Bell System Tech. J. 44, p 2245-2269 (1965).

[3] Lin S. & Kernighan B., "An effective heuristic algorithm for the traveling salesman problem", Op. Res. 21, p.498-516 (1973).

[4] Rosenkrantz D , Sterns R. & Lewis P., "An analysis of several heuristics for the traveling salesman problem", SIAM J. Comp. 6, p. 563-581 (1977).

[5] Svestka J. & Huckfeldt V., "Computational experience with an M-salesmen traveling salesman algorithm", Management Sc. 19, p. 790-799 (1973).